

Homework 10 - OOP

CS 1301 - Intro to Computing - Spring 2022

Important

- Due Date: **Tuesday, April 19th, 11:59 PM.**
- This is an individual assignment. High-level collaboration is encouraged, **but your submission must be uniquely yours.**
- Resources:
 - TA Helpdesk
 - Email TA's or use class Piazza
 - [How to Think Like a Computer Scientist](#)
 - [CS 1301 YouTube Channel](#)
- Comment out or delete all function calls. Only import statements, global variables, and comments are okay to be outside of your functions.
- **Read the entire document before starting this assignment.**

Python is one of many coding languages which uses object oriented programming (OOP). In OOP, classes can be created which contain certain attributes and methods which are shared by all objects of that class. This helps you create concise code which you can re-use. The goal of this homework is to understand OOP and its real world application.

Hidden Test Cases: In an effort to encourage debugging and writing robust code, we will be including hidden test cases on Gradescope for some functions. You will not be able to see the input or output to these cases. Below is an example output from a failed hidden test case:

```
Test failed: False is not true
```

Written by [Arushi Gupta \(arushigupta@gatech.edu\)](mailto:arushigupta@gatech.edu), [Josh Tabb \(jtabb@gatech.edu\)](mailto:jtabb@gatech.edu) & [Assata Quinichett \(aquinichett@gatech.edu\)](mailto:aquinichett@gatech.edu)

Introduction

For this assignment, you will be building classes based on Tik Tok. Tik Tok is a social media application where users can share content in the form of videos. There will be three classes working together: a Video class, an Account class, and a TikTok class.

Each of these classes will have attributes and methods, as described below. You have been provided with a file that has the beginning of these classes. You are responsible for filling in the rest, and the methods you need will be clearly listed out in the grading rubric. **Please read the entire assignment before writing your code in order to understand how all of the classes interact with each other!**

Note: The given `HW10.py` has `__repr__` methods, as well as some other useful methods that we have already defined for you. Do not delete or change these, as these are needed for testing.

Video

Attributes:

- sound (`str`): sound used in the video
- likes (`int`): the number of likes the video has
- comments (`int`): number of all the comments a video has
- minutes (`float`): the length of the video

Methods:

- `__init__`
 - Initializes the following attributes
 - sound (`str`)
 - minutes (`float`) - If no value is provided, the number of minutes should be set to 1.0.
 - likes (`int`) - Every video must begin with 0 likes.
 - comments (`int`) - Every video must begin with 0 comments.
- `__str__`
 - This method does not take in any arguments and creates a string representation of a Video object.
 - The representation of the `Video` object should be in the following format: `"Video has (likes) likes and is (minutes) minutes long."`
- `__eq__`
 - This method checks if two `Video` objects are equal. This is true if and only if the videos have the same sound and an equal number of comments.

Account

Attributes:

- username (str): username of an account
- password (str): password for an account
- followers (list): list of Account objects corresponding to followers of a user
- following (list): list of Account objects corresponding to accounts a user is following
- videos (list): list of Video objects corresponding to videos posted by a user
- isPrivate (bool): states if the account is private or public

Methods:

- __ init __
 - Initializes the following attributes
 - username (str)
 - password (str)
 - followers (list) - Initialized as an empty list
 - following (list) - Initialized as an empty list
 - videos (list) - Initialized as an empty list
 - isPrivate (bool) - If no value is provided, this value should be set to True .
- changePassword
 - This method will be changing the current password of the account.
 - This method takes in two additional parameters:
 - oldPassword (str)
 - newPassword (str)
 - If the oldPassword does **not** match the current password of the Account object, then return 'Invalid password.'
 - Otherwise, change the password of Account to the newPassword passed in.
- isInfluencer
 - In this method, we are determining if an account's user is an influencer or not.
 - This method does not take in any additional arguments
 - To be an influencer, the Account object must have at least 5 videos that satisfy the following requirements:
 - The video has at least 50,000 likes.

- The video has at least 1,000 comments.
 - The account is a public account.
- If all conditions listed above are met, the account's user is an influencer and this method should return `True` . Otherwise, return `False` .
- `follow`
 - This method will take in one additional parameter: `followedAccount (Account)`
 - In this method, add the given `followedAccount` to the current `Account` 's list of following
 - Additionally, the `followedAccount` should also add the current `Account` to its list of followers
 - Note: If the current `Account` is already following `followedAccount`, then do nothing.
- `unfollow`
 - This method should take in one additional parameter: `unfollowedAccount (Account)`
 - This method should remove `unfollowedAccount` from the current `Account` 's following list
 - Additionally, the `unfollowedAccount` should remove the current `Account` from its list of followers
 - If the current `Account` is not already following `unfollowedAccount`, then do nothing.
- `__lt__`
 - One `Account` object is considered less than another if the total number of likes for an account (i.e. the sum of likes on every video posted by an account) is less than the total number of likes of another account
- `__eq__`
 - Two `Account` objects are equal if and only if they have the same `username` and `password` attributes

TikTok

Attributes:

- `accounts (list)`: a list of registered `Account` objects
- `followers (list)`: a list of tuples where each tuple is formatted as: `(Account username, Account followers)`
- `following (list)`: a list of tuples where each tuple is formatted as: `(Account username, Account following)`
- `videos (list)`: a list of `video` objects that are in the application
- `soundBase (list)`: a list of sounds `(str)` that are in the application

Methods:

- `__init__` (provided in `HW10.py`)
 - Initializes the following attributes
 - `accounts (list)`: A list of accounts on `TikTok` that is empty upon initialization.
 - `followers (list)`: The followers list should be empty upon initialization.
 - `following (list)`: The following list should be empty upon initialization.
 - `videos (list)`: A list of all `video` objects that have been posted. This list should be empty upon initialization.
 - `soundBase (list)`: A list of sounds in `TikTok`'s base. This list will also be empty upon initialization.
- `makeAccount`
 - In this method, a new `TikTok` account is being registered.
 - This method should take in two additional parameters:
 - `username (str)`
 - `password (str)`
 - This method should first check if the given username already exists in `TikTok`'s current list of accounts.
 - If the username already exists:
 - Return `"Username is already taken."`
 - If the username does not exist:
 - Create a new `Account` object with the given username and password
 - Add the new `Account` object to `TikTok`'s list of accounts
 - Update `TikTok`'s followers list by adding a new tuple of the `(Account username, Account followers)`

- Update TikTok's following list by adding a new tuple of the (Account username, Account following)
- changePrivacy
 - In this method, change the privacy flag of an account on TikTok .
 - This method takes in one additional parameter: user (Account)
 - In this method, if the user currently has a private account, the user's account should then be public.
 - If the user currently has a public account, the user's account will then be private.
- deleteAccount
 - In this method, a current TikTok account is being deleted.
 - This method takes in one additional parameter: user (Account)
 - If the given account does not already exist in TikTok's list of registered accounts, then do nothing.
 - Otherwise, this method should:
 - Remove the user from the list of accounts of the current TikTok object
 - Remove the tuple pertaining to the user from the current TikTok object's following and follower lists
- post
 - This method takes in three additional parameters:
 - user (Account)
 - sound (str)
 - minutes (float)
 - If the given user account is not registered (not contained in the list of accounts of the current TikTok object), return 'You must have a registered account to post.'
 - If the length of the video is less than 15 seconds (note that the length was given to you in **minutes**), return 'Video must be at least 15 seconds.'
 - Otherwise, create a new Video object with the given parameters.
 - The new Video object should be added to the list of videos of the user account
 - The new Video object should also be added to the list of videos of the current TikTok object
- getInfluencers
 - This method takes in no additional parameters
 - This method should return a list of Account objects that are influencers from the list of accounts in the current TikTok object

- Hint: The `isInfluencer` method in the `Account` class will be helpful for this method.

Grading Rubric

Class: Method	Points
Video: __ init __	4
Video: __ str __	4
Video: __ eq __	4
Account: __ init __	4
Account: changePassword	8
Account: isInfluencer	8
Account: follow	10
Account: unfollow	10
Account: __ lt __	4
Account: __ eq __	4
TikTok: makeAccount	11
TikTok: changePrivacy	5
TikTok: deleteAccount	11
TikTok: post	8
TikTok: getInfluencers	5
Total	100

Provided

The `HW10.py` skeleton file has been provided to you. This is the file you will edit and implement. All instructions for what the functions should do are in this skeleton and this document.

Submission Process

For this homework, we will be using Gradescope for submissions and automatic grading. When you submit your `HW10.py` file to the appropriate assignment on Gradescope, the autograder will run automatically. The grade you see on Gradescope will be the grade you get, unless your grading TA sees signs of you trying to defeat the system in your code. You can re-submit this assignment an unlimited number of times until the deadline; just click the “Resubmit” button at the lower right-hand corner of Gradescope. You do not need to submit your `HW10.py` on Canvas.